

This Page Is Inserted by IFW Operations  
and is not a part of the Official Record

## **BEST AVAILABLE IMAGES**

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images may include (but are not limited to):

- BLACK BORDERS
- TEXT CUT OFF AT TOP, BOTTOM OR SIDES
- FADED TEXT
- ILLEGIBLE TEXT
- SKEWED/SLANTED IMAGES
- COLORED PHOTOS
- BLACK OR VERY BLACK AND WHITE DARK PHOTOS
- GRAY SCALE DOCUMENTS

**IMAGES ARE BEST AVAILABLE COPY.**

**As rescanning documents *will not* correct images,  
please do not report the images to the  
Image Problem Mailbox.**

(19)



JAPANESE PATENT OFFICE

PATENT ABSTRACTS OF JAPAN

(11) Publication number: **07297725 A**

(43) Date of publication of application: **10 . 11 . 95**

(51) Int. Cl. **H03M 7/30**  
**G10L 7/04**  
**H03H 17/02**  
**H04B 1/66**  
**H04B 14/04**  
**H04N 7/30**

(21) Application number: **06082831**

(22) Date of filing: **21 . 04 . 94**

(71) Applicant: **FUJITSU LTD**

(72) Inventor: **YAMASHITA KIMIAKI**  
**HAYASHI HIDEYUKI**  
**MATSUO NAOJI**  
**INAMOTO YASUSHI**  
**MATSUDA KIICHI**

(54) **BAND SYNTHESIS FILTER**

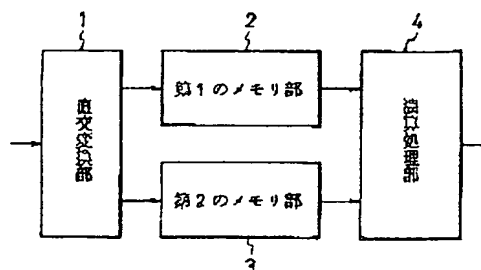
(57) Abstract:

**PURPOSE:** To reduce the arithmetic quantity of the band synthesis filter by providing an orthogonal transformation section generating a 2N-degree vector from N-sets of division data, 1st and 2nd memory sections generating a 2MN- degree vector and an arithmetic processing section implementing band synthesis to the filter.

**CONSTITUTION:** An orthogonal transformation section 1 applies orthogonal transformation to N-sample data to generate a 2N-degree vector. The first half part 0-N-1 of the 2N-degree vector and the latter half part N-2N-1 are used for a block unit of the processing and when the first half part is stored in the 1st memory section 2, the latter half part is stored in the 2nd memory section 3. Then an arithmetic processing section 4 implements the processing of band synthesis by using the 2MN-degree vector stored in the 1st memory section 2 storing the first half part. Then the latter half part of the block unit is stored in the 1st memory section 2 and the first half part is stored in the 2nd memory section 3 and the arithmetic processing section 4 implements the processing of band synthesis by using the 2MN-degree vector stored in the 1st memory section 3.

Thus, the arithmetic quantity for the filter processing and the memory using quantity are reduced.

COPYRIGHT: (C)1995,JPO



(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平7-297725

(43) 公開日 平成7年(1995)11月10日

(51) Int. Cl. <sup>6</sup>	識別記号	庁内整理番号	F I	技術表示箇所
H 0 3 M 7/30	A	8842-5 J		
G 1 0 L 7/04	G			
H 0 3 H 17/02	B	8842-5 J		
H 0 4 B 1/66				

H 0 4 N 7/ 133

Z

審査請求 未請求 請求項の数 1 O L (全 6 頁) 最終頁に続く

(21) 出願番号 特願平6-82831

(22) 出願日 平成6年(1994)4月21日

(71) 出願人 000005223

富士通株式会社

神奈川県川崎市中原区上小田中1015番地

(72) 発明者 山下 公彰

福岡県福岡市博多区博多駅前三丁目22番8号 富士通九州ディジタル・テクノロジー株式会社内

(72) 発明者 林 秀之

福岡県福岡市博多区博多駅前三丁目22番8号 富士通九州ディジタル・テクノロジー株式会社内

(74) 代理人 弁理士 柏谷 昭司 (外1名)

最終頁に続く

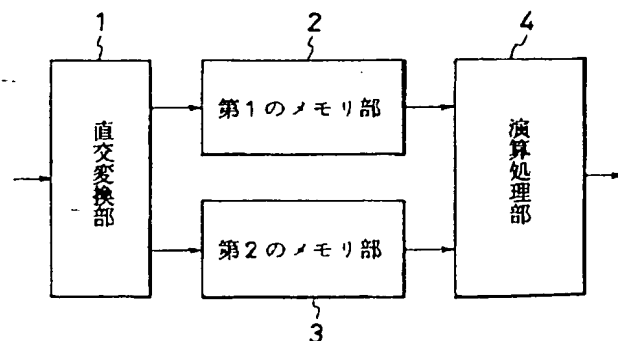
(54) 【発明の名称】 帯域合成フィルタ

(57) 【要約】

【目的】 帯域分割符号化信号を合成する帯域合成フィルタに関し、演算量を削減する。

【構成】 N帯域に分割されたN個のデータを直交変換して2N次ベクトルを生成する直交変換部1と、この直交変換部1により変換された2N次ベクトルを処理のブロック単位とし、そのブロック単位の前半部と後半部とを交互に切替えてストアし、2MN次ベクトルを生成する第1、第2のメモリ部2、3と、現ブロック単位の前半部をストアしたメモリ部に於ける2MN次のベクトルを用いて帯域合成を行う演算処理部4とを備えている。

本発明の実施例の説明図



## 【特許請求の範囲】

【請求項1】 N帯域に分割されたデータを入力して帯域合成する帯域合成フィルタに於いて、

前記N帯域に分割されたN個のデータを直交変換して2N次のベクトルを生成する直交変換部(1)と、

該直交変換部(1)により変換された2N次のベクトルを処理のブロック単位とし、該ブロック単位の前半部と後半部とを交互に切替えてストアしてそれぞれ2MN次のベクトルを生成する第1、第2のメモリ部(2)、

(3)と、

該第1、第2のメモリ部(2)、(3)の何れか一方の現ブロックの前半部をストアしたメモリ部に於ける2MN次ベクトルを用いて帯域合成を行う演算処理部(4)とを備えたことを特徴とする帯域合成フィルタ。

## 【発明の詳細な説明】

## 【0001】

【産業上の利用分野】本発明は、帯域分割符号化された信号を合成処理する帯域合成フィルタに関する。音声信号や画像信号等を周波数軸上で帯域分割して処理する各種の方式が知られている。例えば、周波数多重化信号の生成、分離、又は秘話通信の為の周波数スクランブル、又は音声、画像信号のサブバンド符号化等がある。このような帯域分割、合成の手段として、ポリフェーズフィルタバンクが知られており、帯域合成の場合の処理の効率化が要望されている。

## 【0002】

【従来の技術】音声信号や画像信号等の圧縮符号化方式の一つとして、例えば、国際標準のMPEG(Moving Picture Image Coding Experts Group)が知られている。このMPEGによる音声信号の符号化は、32帯域に分割し、且つ聴覚特性を利用して適応的なビット割当てを行うものである。

【0003】図4はMPEGオーディオ符号化方式の説明図であり、30は符号化部、40は復号化部、31はマッピング部、32は量子化・符号化部、33は聴覚モデルによる重み付け部、34はビット多重化部、35はビット分離部、36は逆量子化・復号化部、37は逆マッピング部である。

【0004】入力信号は、32kHz、41.1kHz、48kHzの何れかのサンプリング周波数によってサンプリングされてマッピング部31に入力され、この時間領域のサンプルは、64タップのポリフェーズフィルタバンクにより32帯域に分割される。その場合、32サンプルを入力して、480サンプル分のオーバーラップを含めて512サンプルのブロックを形成し、その512サンプルのブロックについて窓関数を乗算した後、前述の64タップのポリフェーズフィルタバンクによる帯域分割処理を行い、32サブバンド出力を得ることができる。

【0005】量子化・符号化部32は、32帯域毎の最

大値検索により、量子化をフルスケールで行う為のスケールファクターを求め、このスケールファクターによる振幅正規化を、各帯域で12サンプル毎に行う。又スケールファクターの符号化は6ビットのスケールファクターテーブルを用いる。

【0006】又聴覚モデルによる重み付け部33は、聴覚マスキング効果による重み付けを行うもので、例えば、各サブバンド信号のエネルギーを計算し、この値に最小可聴限の閾値を加算してマスキング閾値とし、このマスキング閾値とサブバンド信号のピークエネルギーとを比較し、マスキング閾値より低い場合はそのサブバンド信号を送送する必要がないので、量子化・符号化部32に通知する。

【0007】ビット多重化部34は、サブバンドの符号化データや補助データ等を多重化し、384サンプル単位でフレーム化し、この符号化信号を送送或いは蓄積する。その場合、例えば、レイヤIのフレーム構成は、16ビットの同期用ビットと、4ビットのビットレート情報と、2ビットのサンプリング周波数情報と、2ビットの符号化モードと、2ビットの著作権の有無情報と、2ビットのエンフェシスの有無情報と、8ビットのビット配分情報と、12ビットのスケールファクターと、帯域分割符号化された符号化ビットと、ダミービットとを含むものである。

【0008】符号化信号の受信側又は再生側に於ける復号化部40は、符号化部30の逆の処理を行うものであり、符号化部30の符号化信号をビット分離部35に於いて補助データ等とサブバンド符号化データとを分離し、逆量子化・復号化部36に於いてサブバンド対応に復号処理する。又逆マッピング部37は、例えば、マッピング部31に於けるポリフェーズフィルタバンクと同様なポリフェーズフィルタバンクを備えて、帯域合成を行い、再生出力信号とするものである(MPEGオーディオ符号化方式は、例えば、1992年10月29日産業図書株式会社発行、白井克彦監修 海上重之、雁部洋久、池沢斗志、松村俊彦、天野文雄 著「ディジタル信号処理の応用」第76頁、第77頁参照)。

【0009】図5は従来例の帯域合成フィルタバンクのフローチャートであり、前述のMPEGオーディオ符号化方式の逆マッピング部に相当する処理の要部を示す。即ち、ステップ(a)～(f)内で示す演算処理を行うものである。まず、Nサンプル分のデータ $S[i]$ ( $i=0, 1, 2, \dots, N-1$ )を入力する(a)。又前回の0～4MN-1のベクトル $V_i$ を2Nサンプル分だけシフトする(b)。又ステップ(a)に於いて入力されたNサンプルの入力信号をモデファイド離散コサイン変換(MDCT)により直交変換して0～2N-1のベクトルを生成し(c)、2Nサンプル分シフトしたベクトル $V_i$ の先頭に付加して、今回の0～4MN-1のベクトルとする。なお、\*は乗算記号であり、SkはNサ

3

ンブル  $S[i]$  に相当する、

【0010】次にベクトル  $U_i$  を生成する (d)。このステップは、C 言語表現による場合を示し、直交変換により求めた  $2N$  次ベクトルを、 $2N$  サンプル分シフトした前回のベクトル  $V_i$  の先頭に付加して、 $4MN-1$  サンプル分の新たなベクトル  $V_i$  を生成し、各  $2N$  サンプル分の前半の  $0 \sim N-1$  と後半の  $N \sim 2N-1$  とを選択的に抽出してベクトル  $U_i$  を生成する。

【0011】このベクトル  $U_i$  に窓関数  $D_i$  を乗算してベクトル  $W_i$  を生成する (e)。次に、このベクトル  $W_i$  の畳み込み演算等により帯域合成処理を行う (f)。この合成出力信号  $S_j$  をアナログ信号に変換することにより、入力信号と同じ出力信号が得られる。

【0012】図6は従来例のベクトル生成の説明図であり、図5のステップ (a) に相当する  $0 \sim N-1$  の  $N$  サンプル  $S_i$  を入力し、直交変換処理を行う。それによって、 $0 \sim 2N-1$  の  $2N$  次のベクトルが得られる。又メモリ等に格納された更新前のベクトル  $V_i$  を  $2N$  サンプル分シフトし、その先頭に新たな  $2N$  サンプル分の  $2N$  次ベクトルを付加する。それによって、前回の  $4MN$  サンプルのベクトル  $V_i$  が更新されることになる。

【0013】この更新されたベクトル  $V_i$  に於いて、それぞれの  $2N$  次ベクトルの前半  $0 \sim N-1$  と後半  $N \sim 2N-1$  とについて、斜線を施して示すように交互に前半と後半とを抽出して、 $0 \sim 2N-1$  のベクトル  $U_i$  を生成する。このベクトル  $U_i$  に窓関数  $D_i$  を乗算してベクトル  $W_i$  を生成し、このベクトル  $W_i$  を基に帯域合成した信号  $S_j$  を得ることができる。

【0014】次に、新たな  $2N$  サンプル分の  $2N$  次ベクトルが直交変換によって得られると、前回の  $4MN$  サンプルのベクトル  $V_i$  を更新し、前回、それぞれの  $2N$  次ベクトルの前半を抽出した場合はその後半を、又後半を抽出した場合はその前半をそれぞれ抽出して、次のベクトル  $U_i$  を生成し、そのベクトル  $U_i$  に窓関数  $D_i$  を乗算してベクトル  $W_i$  を生成し、このベクトル  $W_i$  の畳み込み処理により帯域合成して信号  $S_j$  を得るもので、これを順次繰り返すことになる。

【0015】

【発明が解決しようとする課題】帯域分割符号化方式に於ける復号化部は、前述のような帯域合成フィルタ処理を行うものであり、この帯域合成フィルタ処理をプロセッサを用いてリアルタイムで実行する場合、演算量を如何に少なくするかが重要となる。前述の従来例の処理に於いては、 $N$  サンプルのデータを入力する毎に直交変換してベクトル  $V_i$  を更新し、そのベクトル  $V_i$  からベクトル  $U_i$  を生成し、そのベクトル  $U_i$  に対して窓関数  $D_i$  を乗算してベクトル  $W_i$  を生成するもので、ベクトル  $V_i$  からベクトル  $U_i$  を形成する処理は比較的複雑であり、帯域合成フィルタ処理の演算量の増大の一因となっている。本発明は、帯域合成フィルタの演算量を削減す

4

ることを目的とする。

【0016】

【課題を解決するための手段】本発明の帯域合成フィルタは、図1を参照して説明すると、 $N$  帯域に分割された  $N$  個のデータを直交変換して  $2N$  次のベクトルを生成する直交変換部1と、この直交変換部1により変換された  $2N$  次のベクトルを処理のブロック単位とし、このブロック単位の前半部と後半部とを交互に切替えてストアし、それぞれ  $2MN$  次のベクトルを生成する第1、第2のメモリ部2、3と、この第1、第2のメモリ部2、3の何れか一方の現ブロックの前半部をストアしたメモリ部に於ける  $2MN$  次ベクトルを用いて帯域合成を行う演算処理部4とを備えている。

【0017】

【作用】直交変換部1は、 $N$  サンプルのデータを直交変換して  $2N$  次ベクトルとし、この  $2N$  次ベクトルの前半部  $0 \sim N-1$  と後半部  $N \sim 2N-1$  とを処理のブロック単位とし、第1のメモリ部2に前半部をストアした時は、後半部を第2のメモリ部3にストアし、且つ前半部をストアした第1のメモリ部2にストアされた  $2MN$  次ベクトルを用いて演算処理部4により帯域合成の処理を行い、次に、第1のメモリ部2にはブロック単位の後半部をストアし、第2のメモリ部3に前半部をストアし、この第2のメモリ部3にストアされた  $2MN$  次ベクトルを用いて演算処理部4により帯域合成の処理を行うものである。

【0018】

【実施例】図1に於ける直交変換部1は、離散コサイン変換 (DCT) 又はモデファイド離散コサイン変換 (MDCT) 等により、 $N$  サンプルのデータを直交変換して、 $2N$  次ベクトルを生成する。その  $2N$  次ベクトルの前半部  $0 \sim N-1$  と後半部  $N \sim 2N-1$  とを処理のブロック単位とする。

【0019】又第1、第2のメモリ部2、3は、ランダムアクセスメモリ (RAM) やシフトレジスタ等によって構成することができ、直交変換処理毎に、 $N$  次ベクトル分だけシフトして今回のブロック単位の前半部をストアしたメモリ部は、次回はブロック単位の後半部をストアする。従って、第1、第2のメモリ部2、3には、ブロック単位の前半部と後半部とが交互に切替えられてストアされて、 $2MN$  次ベクトルを生成していることになる。

【0020】演算処理部4は、ブロック単位の前半部をストアしたメモリ部の  $2MN$  次ベクトルを用いて帯域合成処理を行うものである。この演算処理部4と直交変換部1とは、プロセッサの演算機能によって実現することができる。

【0021】図2は本発明の実施例のフローチャートであり、 $N$  サンプル分のデータ  $S[i]$  ( $i=0, 1, 2, \dots, N-1$ ) を入力し①、第1、第2のメモリ部

10

20

30

40

50

5

2、3にストアされている $2MN$ 次のベクトル $U1i$ 、 $U2i$ をそれぞれ $N$ サンプル分シフトする②。

【0022】そして、帯域合成処理に於いて、前回、第1のメモリ部2にストアされたベクトル $U1i$ を用いたか否かを判定し③、このベクトル $U1i$ を用いた場合は、ステップ④に示す直交変換を行い、第2のメモリ部3に $2N$ 次ベクトルのブロック単位の前半部をストアし、そのベクトル $U2i$ をベクトル $Wi$ として窓関数 $D_i$ を乗算し⑤、帯域合成処理を行う⑥。

【0023】又前回、第2のメモリ部3にストアされたベクトル $U2i$ を用いた場合は、ステップ⑥に示す直交変換を行い、第1のメモリ部2に $2N$ 次ベクトルのブロック単位の前半部をストアし、そのベクトル $U1i$ をベクトル $Wi$ として窓関数 $D_i$ を乗算し⑦、帯域合成処理を行う⑧。なお、ステップ④、⑥は同一の直交変換処理であり、ステップ③の判断結果に対応して直交変換処理結果を選択することができる。

【0024】図3は本発明の実施例のベクトル生成の説明図であり、 $0 \sim N-1$ の $N$ サンプル分のデータ $S[i]$ を入力し、直交変換して $2N$ 次ベクトルとし、その $2N$ 次ベクトルを処理のブロック単位とし、 $0 \sim N-1$ の前半部 $a$ と、 $N \sim 2N-1$ の後半部 $b$ とについて、実線矢印で示すように、前半部 $a$ を第2のメモリ部3にストアして $a_n$ とし、後半部 $b$ を第1のメモリ部2にストアして $b_n$ とした場合、第2のメモリ部3には、 $a_n, b_{n-1}, a_{n-2}, \dots$ からなるベクトル $U1i$ が生成され、第1のメモリ部2には、 $b_n, a_{n-1}, b_{n-2}, \dots$ からなるベクトル $U2i$ が生成される。なお、第1、第2のメモリ部2、3は、ブロック単位の前半部 $a$ 又は後半部 $b$ をストアする前に、 $N$ サンプル分のシフトを行うものである。或いは、第1、第2のメモリ部2、3をシフトレジスタにより構成し、ブロック単位の前半部 $a$ 又は後半部 $b$ をシフトしながら、第1、第2のメモリ部2、3にストアすることができる。

【0025】そして、今回のブロック単位の前半部 $a$ をストアした第2のメモリ部3のベクトル $U2i$ を用いて、このベクトル $U2i$ に窓関数 $D_i$ を乗算してベクトル $Wi$ を生成し、このベクトル $Wi$ を基に帯域合成する。

【0026】次の直交変換による $2N$ 次ベクトルの前半部 $a$ を、点線矢印で示すように、第1のメモリ部2にス

6

トアして $a_{n+1}$ とし、後半部 $b$ を点線矢印で示すように、第2のメモリ部3にストアして $b_{n+1}$ とする。従って、ブロック単位の前半部 $a$ をストアした第1のメモリ部2には、 $a_{n+1}, b_n, a_{n-1}, b_{n-2}, \dots$ からなるベクトル $U1i$ が生成されることになり、今回のブロック単位の前半部 $a$ をストアして形成されたベクトル $U1i$ を用い、このベクトル $U1i$ に窓関数 $D_i$ を乗算してベクトル $Wi$ を生成し、このベクトル $Wi$ の畳み込み処理によって帯域合成し、信号 $S[j]$ を出力する。前述の動作を繰り返して帯域分割符号化データの帯域合成を行うことができる。

【0027】

【発明の効果】以上説明したように、本発明は、直交変換部1により変換された $2N$ 次のベクトルを処理のブロック単位とし、その前半部を第1のメモリ部2にストアとした時は、後半部を第2のメモリ部3にストアし、今回は、ブロック単位の後半部を第1のメモリ部2にストアし、前半部を第2のメモリ部3にストアするように、交互に切替えてストアし、今回前半部をストアしたメモリ部に於ける $2MN$ 次ベクトルを用いて演算処理部4により帯域合成を行うものであり、ベクトル $U1i, U2i$ が形成されるから、従来例のように帯域合成処理の度にベクトル $Vi$ からベクトル $Ui$ を生成する処理とメモリ使用とを省略することができる。従って、帯域合成フィルタ処理の演算量とメモリ使用量とを削減できる利点がある。

【図面の簡単な説明】

【図1】本発明の実施例の説明図である。

【図2】本発明の実施例のフローチャートである。

【図3】本発明の実施例のベクトル生成の説明図である。

【図4】MPEGオーディオ符号化方式の説明図である。

【図5】従来例の帯域合成フィルタバンクのフローチャートである。

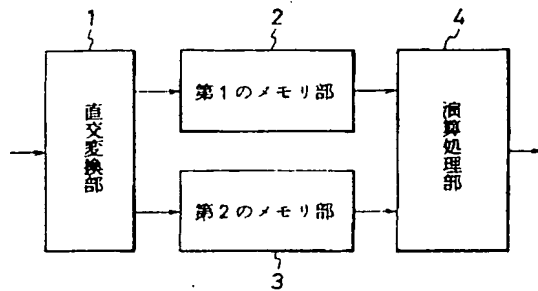
【図6】従来例のベクトル生成の説明図である。

【符号の説明】

- 1 直交変換部
- 2 第1のメモリ部
- 3 第2のメモリ部
- 4 演算処理部

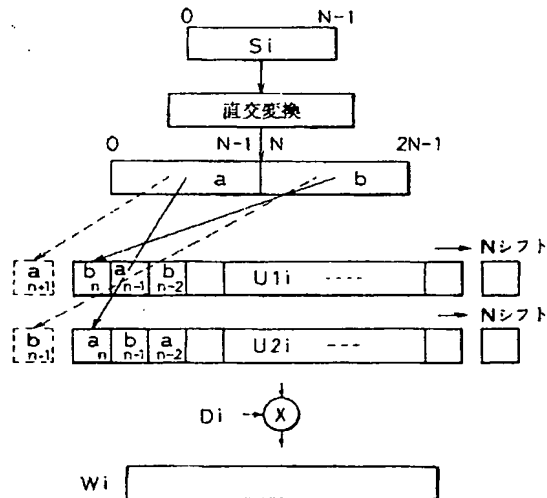
【図1】

本発明の実施例の説明図



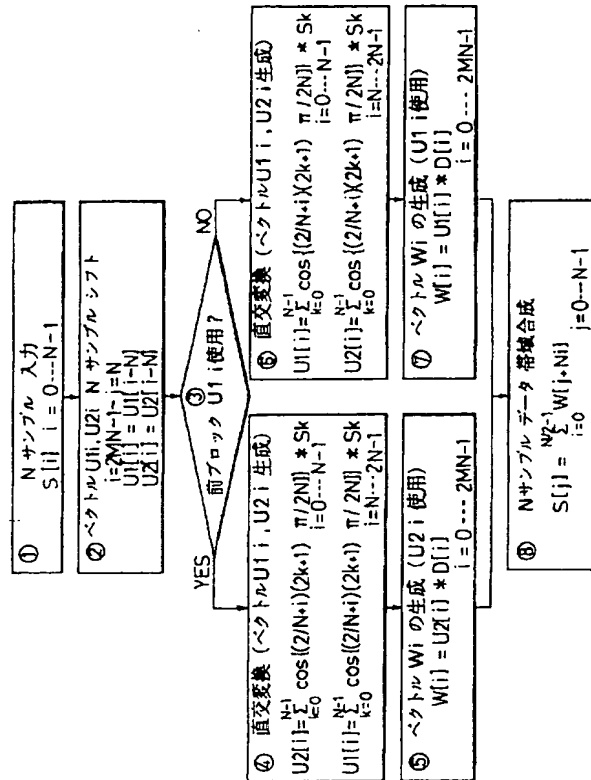
【図3】

本発明の実施例のベクトル生成の説明図



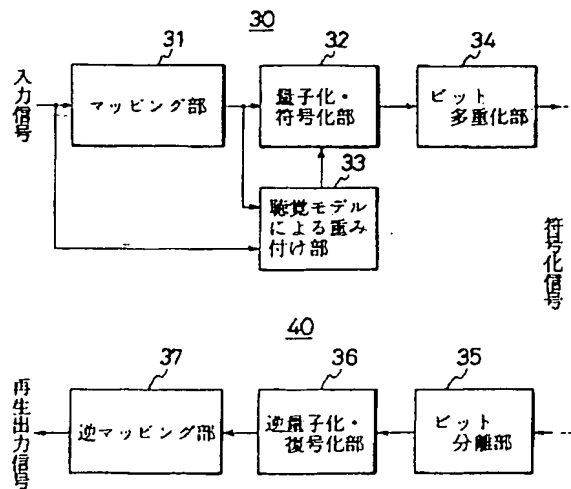
【図2】

本発明の実施例のフローチャート



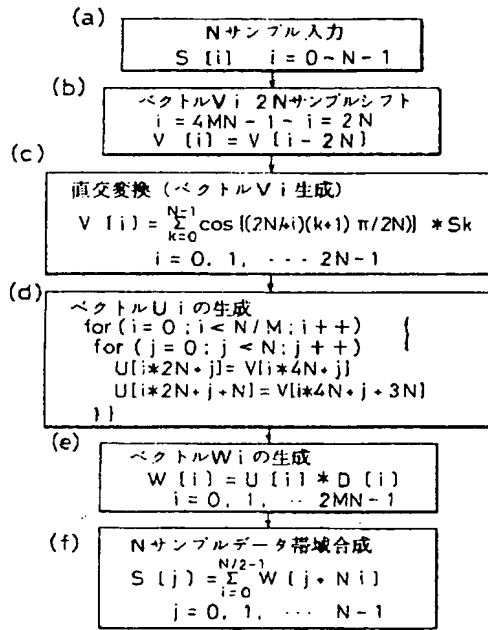
【図4】

MPEGオーディオ符号化方式の説明図



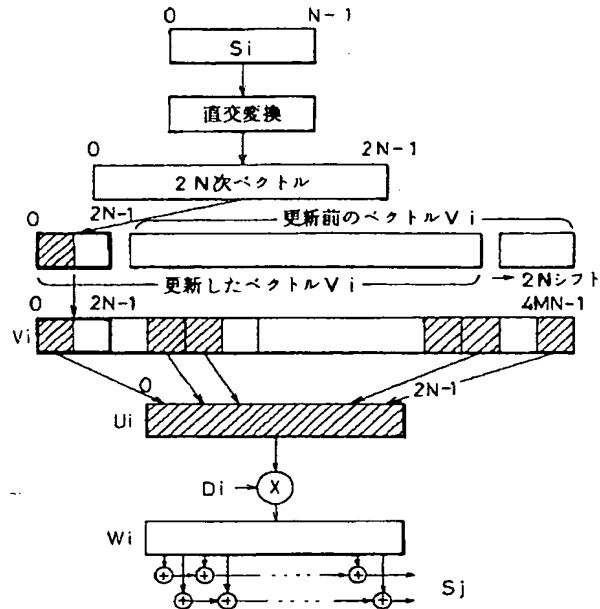
【図5】

従来例の帯域合成フィルタバンクのフローチャート



【図6】

従来例のベクトル生成の説明図



フロントページの続き

(51)Int. Cl.<sup>6</sup>

H 0 4 B 14/04

H 0 4 N 7/30

識別記号

庁内整理番号

Z

F I

技術表示箇所

(72)発明者 松尾 直司

神奈川県川崎市中原区上小田中1015番地  
富士通株式会社内

(72)発明者 稲本 康

神奈川県川崎市中原区上小田中1015番地  
富士通株式会社内

(72)発明者 松田 喜一

神奈川県川崎市中原区上小田中1015番地  
富士通株式会社内



JAPANESE [JP,07-297725,A]

---

CLAIMS DETAILED DESCRIPTION TECHNICAL FIELD PRIOR ART EFFECT OF THE  
INVENTION TECHNICAL PROBLEM MEANS OPERATION EXAMPLE DESCRIPTION OF  
DRAWINGS DRAWINGS

---

[Translation done.]

## \* NOTICES \*

Japan Patent Office is not responsible for any damages caused by the use of this translation.

1. This document has been translated by computer. So the translation may not reflect the original precisely.
2. \*\*\*\* shows the word which can not be translated.
3. In the drawings, any words are not translated.

---

## DETAILED DESCRIPTION

---

### [Detailed Description of the Invention]

[0001]

[Field of the Invention] this invention relates to the band synthesis VCF which carries out synthetic processing of the signal by which band split coding was carried out. Various kinds of methods which carry out the band split of a sound signal, the picture signal, etc., and process them on a frequency shaft are learned. For example, there is sub band coding of the frequency scramble for generation of a frequency multiplexed signal, a separation, or an unknown episode communication or voice, and a picture signal etc. As a means of such a band split and synthesis, the poliphase filter bank is known and the increase in efficiency of processing in the case of being band synthesis is demanded.

[0002]

[Description of the Prior Art] As one of compression coding methods, such as a sound signal and a picture signal, MPEG (Moving Picture Image Coding Experts Group) standard [ for example, ] for international is known. Coding of the sound signal by this MPEG is divided into 32 bands, and performs adaptation bit allocation using an acoustic-sense property.

[0003] For the weighting section [ according / drawing 4 is explanatory drawing of MPEG audio coding method, and / 30 / 40 / the coding section and / 33 / to an acoustic-sense model in the decryption section and 31 ] according [ the mapping section and 32 ] to quantization / coding section, and 34, as for the bit separation section and 36, the bit-interleaved-multiplexing-ized section and 35 are [ reverse quantization / decryption section and 37 ] the reverse mapping sections.

[0004] An input signal is sampled by the sampling frequency of 32kHz, 41.1kHz, or 48kHz, it is inputted into the mapping section 31, and the sample of this time domain is divided into 32 bands by the poliphase filter bank of 64 taps. In this case, after inputting 32 samples, forming a block of 512 samples including the overlap for 480 samples and carrying out the multiplication of the windowing function about a block of the 512 samples, band split processing by the poliphase filter bank of the 64 above-mentioned taps can be performed, and 32 sub band output can be obtained.

[0005] Quantization / coding section 32 asks for the scale factor for quantizing by the full scale by maximum reference for every 32 bands, and performs the amplitude normalization by this scale factor every 12 samples in each band. Moreover, coding of a scale factor uses a 6-bit scale-factor table.

[0006] Moreover, since the weighting section 33 by the acoustic-sense model performs the weighting by the acoustic-sense masking effect, calculates the energy of each sub band signal, for example, adds the threshold of the minimum \*\*\*\*\* to this value, considers as a masking threshold, compares this masking threshold with the peak energy of a sub band signal, and it does not need to transmit the sub band signal when lower than a masking threshold, it is notified to quantization / coding section 32.

[0007] The bit-interleaved-multiplexing-ized section 34 multiplexes coded data, an auxiliary data, etc. of a sub band, frame-izes them per 384 samples, and transmits or accumulates this coded signal. In this case, for example, layer I A frame structure contains the 16 bits bit for a synchronization, a 4 bits bit rate information, a 2 bits sampling-frequency information, the 2-bit

coding mode, the existence information on a 2-bit copyright, the existence information on 2-bit emphasis, a 8 bits bit distribution information, a 12-bit scale factor, the coding bit by which band split coding was carried out, and a dummy bit.

[0008] The receiving side of a coded signal or the decryption section 40 by the side of regeneration performs reverse processing of the coding section 30, separates an auxiliary data etc. and sub-band-coding data in the bit separation section 35, and carries out decode processing of the coded signal of the coding section 30 in reverse quantization / decryption section 36 at sub band correspondence. The \*\*\*\* mapping section 37 is equipped with the poliphase filter bank in the mapping section 31, and the same poliphase filter bank, performs band synthesis, and is taken as a regeneration output signal (for example, October 29, 1992 industrial books incorporated company issue, Katsuhiko Shirai editorial-supervision marine Shigeyuki, \*\*\*\* Hirohisa, Ikezawa \*\*\*\*, Toshihiko Matsumura, and Fumio Amano work "an application of digital signal processing" refer to the 76th page and the 77th page for MPEG audio coding method).

[0009] Drawing 5 is the flow chart of the band synthesis filter bank of the conventional example, and shows the important section of processing equivalent to the reverse mapping section of the above-mentioned MPEG audio coding method. That is, data processing shown within step (a) - (f) is performed. First, data  $S$  for  $N$  sample  $[i]$  ( $i = 0, 1, 2, \dots, N-1$ ) is inputted (a). Moreover, the last vector  $V_i$  of  $0-4MN-1$  is shifted by  $2N$  sample (b). Moreover, orthogonal transformation of the input signal of  $N$  sample inputted in the step (a) is carried out by the \*\*\*\*\* discrete cosine transform (MDCT), and it adds to the head of the vector  $V_i$  which generated the vector of  $0-2N-1$  and was shifted by (c) and  $2N$  sample, and considers as this vector of  $0-4MN-1$ . In addition, \* is an asterisk and  $S_k$  is equivalent to  $N$  sample  $S[i]$ .

[0010] Next, a vector  $U_i$  is generated (d). This step shows the case where it is based on  $C$  representation, adds  $2N$ th vectors searched for by orthogonal transformation to the head of the last vector  $V_i$  shifted by  $2N$  sample, generates the new vector  $V_i$  for  $4MN-1$  sample, extracts alternatively zero to  $N-1$  of the first half for two  $N$  each sample, and  $N-2N-1$  of the second half, and generates a vector  $U_i$ .

[0011] The multiplication of the windowing function  $D_i$  is carried out to this vector  $U_i$ , and a vector  $W_i$  is generated (e). Next, this vector  $W_i$  collapses and an operation etc. performs band synthesis processing (f). By changing this synthetic output signal  $S_j$  into an analog signal, the same output signal as an input signal is obtained.

[0012] Drawing 6 is explanatory drawing of vector generation of the conventional example, inputs  $N$  sample  $S_i$  of zero to  $N-1$  equivalent to the step (a) of drawing 5, and performs orthogonal transformation processing. The  $2N$  order vector of  $0-2N-1$  is acquired by it. Moreover, the vector  $V_i$  before the update stored in memory etc. is shifted by  $2N$  sample, and  $2N$ th vectors for  $2N$  new sample are added to the head. The vector  $V_i$  of last  $4MN$  sample will be updated by it.

[0013] In this updated vector  $V_i$ , the first half and the second half are extracted by turns so that the oblique line may be given and shown in the first half of each  $2N$ th vector about zero to  $N-1$ , and second half  $N-2N-1$ , and the vector  $U_i$  of  $0-2N-1$  is generated. The multiplication of the windowing function  $D_i$  can be carried out to this vector  $U_i$ , a vector  $W_i$  can be generated, and the band synthesis signal  $S_j$  on the basis of this vector  $W_i$  can be acquired.

[0014] Next, if  $2N$ th vectors for  $2N$  new sample are acquired by orthogonal transformation When the vector  $V_i$  of last  $4MN$  sample is updated, the first half of each  $2N$ th vector is extracted last time and the second half is again extracted for the second half, the first half is extracted, respectively. The following vector  $U_i$  is generated, the multiplication of the windowing function  $D_i$  is carried out to the vector  $U_i$ , a vector  $W_i$  will be generated, this vector  $W_i$  will collapse, band synthesis will be carried out by processing, a signal  $S_j$  will be acquired, and this will be repeated one by one.

[0015]

[Problem(s) to be Solved by the Invention] The decryption section in a band split coding method becomes important [ what the amount of operations is lessened ], when the above band synthesis filtering is performed and it performs this band synthesis filtering on real time using a processor.

Whenever it inputs the data of  $N$  sample in processing of the above-mentioned conventional example, orthogonal transformation is carried out, a vector  $V_i$  is updated, a vector  $U_i$  is generated from the vector  $V_i$ , the multiplication of the windowing function  $D_i$  is carried out to the vector  $U_i$ , a vector  $W_i$  is generated, and processing which forms a vector  $U_i$  from a vector  $V_i$  is comparatively complicated, and serves as the cause of increase of the amount of operations of band synthesis filtering. this invention aims at cutting down the amount of operations of a band synthesis VCF.

[0016]

[Means for Solving the Problem] The orthogonal transformation section 1 which will carry out orthogonal transformation of the  $N$  data divided into  $N$  band, and will generate a  $2N$  order vector if the band synthesis VCF of this invention is explained with reference to drawing 1, The 1st and the 2nd memory section 2 and 3 which consider as the block unit of processing of the  $2N$  order vector changed by this orthogonal transformation section 1, change the section by turns, carry out a store the first portion of this block unit, and the second half, and generate the following vector  $2MN$ s, respectively, either of the this 1st [ the ] and 2nd memory section 2 and 3 -- it has the data-processing section 4 in the memory section which carried out the store of the first portion of one present block which performs band synthesis  $2MN$ s using degree vector

[0017]

[Function] The orthogonal transformation section 1 carries out orthogonal transformation of the data of  $N$  sample, and considers as  $2N$ th vectors. When it considers as the block unit of processing of the first portion zero to  $N-1$  of this  $2N$ th vector, and second half section  $N-2N-1$  and the store of the first portion is carried out to the 1st memory section 2 Band synthesis is processed by the data-processing section 4 using  $2MN$  degree vector by which the store was carried out to the 1st memory section 2 which carried out the store of the section to the 2nd memory section 3 in the second half, and carried out the store of the first portion. Next, it is the thing by which carries out the store of the second half section of a block unit to the 1st memory section 2, and carries out the store of the first portion to the 2nd memory section 3, and the store was carried out to this 2nd memory section 3 and which processes band synthesis by the data-processing section 4  $2MN$ s using degree vector.

[0018]

[Example] By the discrete cosine transform (DCT) or the \*\*\*\*\* discrete cosine transform (MDCT), the orthogonal transformation section 1 in drawing 1 carries out orthogonal transformation of the data of  $N$  sample, and generates  $2N$ th vectors. It considers as the block unit of processing of the first portion zero to  $N-1$  of the  $2N$ th vector, and second half section  $N-2N-1$ .

[0019] A RAM (RAM), a shift register, etc. can constitute \*\*\*\* 1 and the 2nd memory section 2 and 3, and the memory section which shifted by the  $N$ th vector and carried out the store of the first portion of this block unit for every orthogonal transformation processing carries out the store of the second half section of a block unit next time. Therefore, the section will be changed by turns to the first portion of a block unit by the 1st and 2nd memory section 2 and 3 in the second half, a store will be carried out to it, and degree vector will be generated  $2MN$ .

[0020] The data-processing section 4 is a thing of the memory section which carried out the store of the first portion of a block unit which performs band synthesis processing  $2MN$ s using degree vector. This data-processing section 4 and the orthogonal transformation section 1 are realizable with the calculation function of a processor.

[0021] Drawing 2 is \*\* by which is the flow chart of the example of this invention, and inputs data  $S$  for  $N$  sample  $[i]$  ( $i=0, 1, 2, \dots, N-1$ ), and the store is carried out to the \*\*, the 1st, and 2nd memory section 2 and 3 and which shifts the following vector  $U_{1i}$  and  $U_{2i}$  by  $N$  sample  $2MN$ s, respectively.

[0022] And it is \*\* which performs orthogonal transformation shown in step \*\*, carries out the store of the first portion of the block unit of  $2N$ th vectors to the 2nd memory section 3, makes the vector  $U_{2i}$  a vector  $W_i$ , carries out the multiplication of the windowing function  $D_i$ , and performs \*\* and band synthesis processing when it judges whether vector  $U_{1i}$  by which the store was carried out to the 1st memory section 2 was used last time in band synthesis processing and \*\*

and this vector  $U_{1i}$  are used.

[0023] Moreover, it is \*\* which performs orthogonal transformation shown in step \*\*, carries out the store of the first portion of the block unit of  $2N$ th vectors to the 1st memory section 2, makes the vector  $U_{1i}$  a vector  $W_i$ , carries out the multiplication of the windowing function  $D_i$ , and performs \*\* and band synthesis processing when vector  $U_{2i}$  by which the store was carried out to the 2nd memory section 3 is used last time. In addition, step \*\* and \*\* are the same orthogonal transformation processings, and can choose an orthogonal transformation processing result corresponding to the decision result of step \*\*.

[0024] Drawing 3 is explanatory drawing of vector generation of the example of this invention, inputs and carries out orthogonal transformation of the data  $S$  for  $N$  sample of zero to  $N-1$  [ $i$ ], makes it  $2N$ th vectors, and is made into the block unit of processing of the  $2N$ th vector. First portion  $a$  of zero to  $N-1$ , About second half section  $b$  of  $N-2N-1$ , as the solid-line arrow head shows, the store of the first portion  $a$  is carried out to the 2nd memory section 3, and it is  $a_n$ . It carries out. The store of the section  $b$  is carried out to the 1st memory section 2 in the second half, and it is  $b_n$ . When it carries out, among the 2nd memory section 3 Vector  $U_{1i}$  which consists of  $a_n, b_{n-1}, a_{n-2}$ , and ... is generated, and vector  $U_{2i}$  which consists of  $b_n, a_{n-1}, b_{n-2}$ , and ... is generated by the 1st memory section 2. In addition, the 1st and 2nd memory section 2 and 3 shifts a part for  $N$  sample, before carrying out the store of the section  $b$  first portion  $a$  of a block unit, or the second half. Or a store can be carried out to the 1st and 2nd memory section 2 and 3, a shift register constituting the 1st and 2nd memory section 2 and 3, and shifting section  $b$  first portion  $a$  of a block unit, or the second half.

[0025] And using vector  $U_{2i}$  of the 2nd memory section 3 which carried out the store of the first portion  $a$  of this block unit, the multiplication of the windowing function  $D_i$  is carried out to this vector  $U_{2i}$ , a vector  $W_i$  is generated and band synthesis is carried out on the basis of this vector  $W_i$ .

[0026] As the dotted-line arrow head shows, the store of the first portion  $a$  of  $2N$ th vectors by the following orthogonal transformation is carried out to the 1st memory section 2, and it is  $a_{n+1}$ . It carries out, as the dotted-line arrow head shows in the second half, the store of the section  $b$  is carried out to the 2nd memory section 3, and it is  $b_{n+1}$ . It carries out. therefore, among the 1st memory section 2 which carried out the store, first portion  $a$  of a block unit Vector  $U_{1i}$  which consists of  $a_{n+1}, b_n, a_{n-1}, b_{n-2}$ , and ... will be generated. Using vector  $U_{1i}$  which carries out the store of the first portion  $a$  of this block unit, and was formed, the multiplication of the windowing function  $D_i$  is carried out to this vector  $U_{1i}$ , a vector  $W_i$  is generated, this vector  $W_i$  collapses, by processing, band synthesis is carried out and signal  $S[j]$  is outputted. The above-mentioned operation can be repeated and band synthesis of band split coded data can be performed.

[0027]

[Effect of the Invention] As explained above, when this invention is made into the block unit of processing of the  $2N$  order vector changed by the orthogonal transformation section 1 and the first portion is made into a store at the 1st memory section 2 The store of the section is carried out to the 2nd memory section 3 in the second half. next time So that the store of the second half section of a block unit may be carried out to the 1st memory section 2 and the store of the first portion may be carried out to the 2nd memory section 3 It is the thing in the memory section which changed by turns, carried out the store and carried out the store of the first portion this time which performs band synthesis by the data-processing section 4  $2MN$ s using degree vector. Since vector  $U_{1i}$  and  $U_{2i}$  are formed, processing and memory use which generate a vector  $U_i$  from a vector  $V_i$  to the degree of band synthesis processing like the conventional example are omissible. Therefore, there is an advantage which can cut down the amount of operations and the amount of the memory used of the band synthesis filtering used.

[Translation done.]